

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (currently amended): A method for localization of a ~~Java computer software~~ application, comprising:
 - locating a plurality of localizable variables of a class using a custom class loader;
 - finding a corresponding resource file for a current language for each localizable variable by searching variables with predefined prefix;
 - calculating a key for each localizable variable by deleting the prefix and combining a class name and a variable name;
 - finding a localized string in the resource file corresponding to each key; and
 - assigning the localized string to the corresponding localizable variable of the class, wherein the custom class loader provides localization of the class during class loading.
2. (original): The method of claim 1, further comprising:
 - creating an instance of the custom class loader using a launcher;
 - requesting an application startup class from the custom class loader;
 - returning a localized version of the application startup class; and
 - running the application startup class using a reflection application programming interface (API).
3. (original): The method of claim 1, further comprising loading resource files of the class from a storage device using the custom class loader.
4. (currently amended): The method of claim 1, further comprising:
 - loading the class from a bootstrap class loader; and
 - passing a localized class to a ~~Java~~ virtual machine.
5. (currently amended): An apparatus for localization of a ~~Java computer software~~ application, comprising:

a custom class loader capable of inspecting codes of the application, finding a corresponding resource file for a current language for each localizable variable by searching variables with predefined prefix, calculating a key for each localizable variable by deleting the prefix and combining a class name and a variable name, and converting strings of a class to localized strings during class loading, thereby providing localization of the class during class loading;

a reflection application programming interface (API) capable of obtaining descriptors of variables of the class; and

a launcher capable of creating an instance of the custom class loader and running an application startup class using the reflection API.

6. (currently amended): The apparatus of claim 5, wherein the custom class loader is a subclass of a Java language class loader.

7. (original): The apparatus of claim 5, wherein the custom class loader locates the plurality of localizable variables of the class.

8-9. (cancelled).

10. (currently amended): The apparatus of claim 9 7, wherein the custom class loader finds a localized string in the resource file corresponding to each key.

11. (original): The apparatus of claim 10, wherein the custom class loader assigns the localized string to the corresponding localizable variable of the class.

12. (original): The apparatus of claim 5, wherein the custom class loader loads the class from a bootstrap class loader.

13. (currently amended): The apparatus of claim 5, wherein the custom class loader passes a localized class to a Java virtual machine.

14. (original): The apparatus of claim 5, wherein the launcher creates an instance of the custom class loader.

15. (original): The apparatus of claim 5, wherein the launcher requests an application startup class from the custom class loader, wherein the custom class loader returns a localized version of the application startup class.

16. (original): The apparatus of claim 15, wherein the custom class loader runs the application startup class using a reflection application programming interface (API).

17. (currently amended): A computer readable medium providing instructions for localization of a Java computer software application, the instructions comprising:

locating a plurality of localizable variables of a class using a custom class loader;
finding a corresponding resource file for a current language for each localizable variable by searching variables with predefined prefix;
calculating a key for each localizable variable by deleting the prefix and combining a class name and a variable name;
finding a localized string in the resource file corresponding to each key; and
assigning the localized string to the corresponding localizable variable of the class,
wherein the custom class loader provides localization of the class during class loading.

18. (original): The computer readable medium of claim 17, further comprising instructions for:

creating an instance of the custom class loader using a launcher;
requesting an application startup class from the custom class loader;
returning a localized version of the application startup class; and
running the application startup class using a reflection application programming interface (API).

19. (original): The computer readable medium of claim 17, further comprising instructions for loading resource files of the class from a storage device using the custom class loader.

20. (currently amended): The computer readable medium of claim 17, further comprising instructions for:

loading the class from a bootstrap class loader; and

passing a localized class to a Java virtual machine.